



# Installation Guide

Computer Management Sciences, Inc  
6076-D Franconia Road  
Alexandria, Virginia 22310-1756  
voice: (703) 922-7027  
fax: (703) 922-7305  
[www.cpexpert.com](http://www.cpexpert.com)

CPExpert is a trademark of Computer Management Sciences, Inc.

This document provides installation instructions for **CPExpert™**, a proprietary software product of Computer Management Sciences, Inc., Alexandria, Virginia, United States of America.

The information in this document is subject to change. Comments or suggestions are welcome, and to the extent practicable, will be incorporated in revisions to this document. Please send all comments, suggested new rules, suggested changes to existing rules, suggestions for improvement to the software or documentation, or any other advice to:

Computer Management Sciences, Inc.  
6076-D Franconia Road  
Alexandria, Virginia 22310-1756  
(703) 922-7027 FAX: (703) 922-7305  
**[www.cpexpert.com](http://www.cpexpert.com)**

### **DISCLAIMER**

The advice, recommendations, or otherwise contained in this document represent information generally available in the public domain, as contained in vendor manuals, published in articles or papers, presented at professional conferences, or otherwise commonly accepted in the professional community. Neither Computer Management Sciences, Inc. nor its representatives make representations or warranties with respect to the applicability or application of any advice, recommendations, or otherwise, contained in this document or in any results from applying the CPExpert software, to any particular computer system or computer installation.

### **TRADEMARKS**

CPExpert is a trademark of Computer Management Sciences, Inc. IBM, MVS/370, MVS/SP, MVS/XA, MVS/ESA, Enterprise System/3090, Netview, PR/SM, Processor Resource/System Manager, Hiperspace, and ES/9000 are trademarks of the IBM Corporation. SAS, SAS/OR, and SAS/STAT are trademarks of the SAS Institute Inc. MXG is a trademark of Merrill Consultants. All other trademarks are the property of their respective owners.

### **COPYRIGHT INFORMATION**

©Copyright 1991, Computer Management Sciences, Inc.  
All rights reserved. Printed in the United States of America.

This licensed work is confidential and propriety, and is the property of Computer Management Sciences, Inc. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the specific written authorization of Computer Management Sciences, Inc.

# Contents

---

Page

## Section 1: Introduction

## Section 2: Installing CPExpert on a mainframe under MVS

Chapter 1: Unload the distribution tape, using IEBUPDTE .....	2-1
Chapter 2: Allocate space for CPExpert data sets .....	2-3

## Section 3: Optional general guidance to CPExpert

Chapter 1: Type of performance data base .....	3-4
Chapter 2: MVS version .....	3-4
Chapter 3: PDBLIB JCL DD statement .....	3-4
Chapter 4: SMF Type 30 (Job Initiation) records available .....	3-4
Chapter 5: SMF Type 30 (Job Step) records available .....	3-5
Chapter 6: SMF Type 30 (Job) records available .....	3-5
Chapter 7: SMF Type 30 (Interval) records available .....	3-5
Chapter 8: SMF Type 30 modification to MXG or MICS installed .....	3-5
Chapter 9: SMF Type 42 (Data Set) records available .....	3-6
Chapter 10: SMF Type 64 (VSAM Statistics) records available .....	3-6
Chapter 11: SMF Type 88 (System Logger) records available .....	3-6
Chapter 12: Page size for printer output .....	3-7
Chapter 13: SAS printer output destination .....	3-7
Chapter 14: PR/SM, MDF, or MLPF .....	3-7
Chapter 15: CPU service units (page-in) constant .....	3-7
Chapter 16: CPU service units (EXCP) constant .....	3-8
Chapter 17: CPU service units (swap) constant .....	3-8
Chapter 18: MICS processing cycle .....	3-8
Chapter 19: MICS product change level .....	3-8
Chapter 20: MICS variable derivation option .....	3-9
Chapter 21: MICS JCL DD statement options .....	3-10
Chapter 22: MICS MICJOB, MIC_TS, and MIC_ST variables .....	3-11
Chapter 23: MXG JCL DD statement options .....	3-11
Chapter 24: SAS Output Delivery System .....	3-12
Chapter 24.1: SAS Output Delivery System - SASODS variable .....	3-13
Chapter 24.2: SAS Output Delivery System - PATH variable .....	3-15
Chapter 24.3: SAS Output Delivery System - FRAME variable .....	3-16
Chapter 24.4: SAS Output Delivery System - CONTENTS variable .....	3-16
Chapter 24.5: SAS Output Delivery System - BODY variable .....	3-16
Chapter 24.6: SAS Output Delivery System - LINKPDF variable .....	3-17
Chapter 24.7: SAS Output Delivery System - STYLE variable .....	3-18

**Page**

Chapter 24.8: SAS Output Delivery System - PDFODS variable . . . . .	3-18	
Chapter 24.9: SAS Output Delivery System - PDFFILE variable . . . . .	3-18	
Chapter 24.10: SAS Output Delivery System - TRANTAB variable . . . . .	3-18	
Chapter 24.11: SAS Output Delivery System - URL variable . . . . .	3-18	

**Section 4: Files used by CPExpert**

Chapter 1: Files required when processing a MXG PDB . . . . .	2
Chapter 1.1: Files used by the WLM Component . . . . .	4
Chapter 1.2: Files used by the DB2 Component . . . . .	5
Chapter 1.6: Files used by the CICS Component . . . . .	6
Chapter 1.5: Files used by the DASD Component . . . . .	8
Chapter 1.3: Files used by the MVS Component (Compatibility Mode) . . . . .	9
Chapter 1.4: Files used by the TSO Component (Compatibility Mode) . . . . .	10
Chapter 1.2: Files used by the SRM Component (Compatibility Mode) . . . . .	10
Chapter 2: Files required when processing a NeuMICS PDB . . . . .	11
Chapter 2.1: Files used by the WLM Component . . . . .	11
Chapter 2.2: Files used by the DB2 Component . . . . .	12
Chapter 2.3: Files used by the DASD Component . . . . .	13
Chapter 2.4: Files used by the CICS Component . . . . .	13
Chapter 2.5: Files used by the MVS Component (Compatibility Mode) . . . . .	14
Chapter 2.6: Files used by the TSO Component (Compatibility Mode) . . . . .	15
Chapter 2.7: Files used by the SRM Component (Compatibility Mode) . . . . .	15

**Exhibits**

2-1	Job Control Language to install CPExpert under MVS . . . . .	2-1
2-2	Job Control Language to allocate DASD space . . . . .	2-3
3-1	Sample Display of CPEXPRT.USOURCE(GENGUIDE) Module . . . . .	3-2

# Section 1: Introduction

---

This document contains a detailed description of how to install CPEXpert on your system under MVS<sup>1</sup>.

CPEXpert is normally distributed on magnetic tape. The tape was built using IBM's IEBUPDTE utility software.

Section 2 describes how to install CPEXpert. The instructions for installing CPEXpert are very simple, and installation should take only a few minutes of your time. The installation consists of three steps:

- Load the software onto your system.
- Allocate space for the data sets created by CPEXpert.
- Provide general guidance about your environment (contained in the CPEXPERT.USOURCE(GENGUIDE) Module).

Section 3 describes how to modify the CPEXPERT.USOURCE(GENGUIDE) module to provide CPEXpert with information about your environment. **Most parameters in this module are optional.** You will need to modify this module before you execute CPEXpert only if the default values do not accurately describe your environment.

Section 4 describes the files used by different components of CPEXpert, depending on whether the performance data base is MXG or NeuMICS.

Please call Computer Management Sciences if you have **any** problems or questions during installation. We would much rather have you call for information than have you be frustrated by some problem!

---

<sup>1</sup>Please contact Computer Management Sciences if you wish instructions on how to install CPEXpert under CMS or install CPEXpert on a personal computer.



# Section 2: Installing CPEXpert under MVS

---

CPEXpert normally is distributed on a non-labeled tape containing two source format libraries: SOURCE and USOURCE. The tape was created using IBM's IEBUPDTE utility software. This section describes how to install CPEXpert on a mainframe system operating under MVS.

## Chapter 1: Unload the distribution tape, using IEBUPDTE.

The first step is to unload the distribution tape containing the CPEXpert code. The PDS requires fifteen to fifty cylinders of IBM-3380 space (depending upon how many components of CPEXpert you have ordered), and consists of members containing standard 80-byte record.

Use the Job Control Language (JCL) shown in Exhibit 2-1 to unload the data from tape to disk. This JCL unloads the majority of the CPEXpert source code into CPEXpert's normal source library (titled SOURCE), and unloads some CPEXpert source code into a user source library (titled USOURCE).

---

//jobname	JOB	job card information
//STEP01	EXEC	PGM=IEBUPDTE,PARM=NEW
//SYSPRINT	DD	SYSOUT=*
//SYSIN	DD	DSN=CPEXPERT.SOURCE,DISP=(OLD,KEEP),UNIT=TAPE9,
//		VOL=(,RETAIN,SER=CPE000),LABEL=(1,NL,EXPDT=98000),
//		DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUT2	DD	DSN=prefix.CPEXPERT.SOURCE,DISP=(,CATLG,DELETE),
//		UNIT=3380,VOL=SER=volser,SPACE=(CYL,(15,10,40)),
//		DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//STEP02	EXEC	PGM=IEBUPDTE,PARM=NEW
//SYSPRINT	DD	SYSOUT=*
//SYSIN	DD	DSN=CPEXPERT.USOURCE,DISP=(OLD,KEEP),UNIT=TAPE9,
//		VOL=SER=CPE000,LABEL=(2,NL,EXPDT=98000),
//		DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//SYSUT2	DD	DSN=prefix.CPEXPERT.USOURCE,DISP=(,CATLG,DELETE),
//		UNIT=3380,VOL=SER=volser,SPACE=(TRK,(2,1,10)),
//		DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)

---

## JOB CONTROL LANGUAGE TO INSTALL CPEXpert UNDER MVS

### EXHIBIT 2-1

The JCL is only a pattern. You will need to make the following changes:

JOBNAME	JOB	The job card information must be completed according to your installation's requirements
SYSIN	DD	The VOLSER CPEnnnn should be replaced by the actual VOLSER printed on the tape you receive.
SYSUT2	DD	The VOLSER=SER=volser must be replaced by the DASD volume on which you plan to unload the SOURCE. You may also need to change the unit type to SYSDA, based on your installation standards. Additionally, change the prefix in accordance with your installation standards.

Please use data set names (DSN) with USOURCE and SOURCE as the final qualifier to facilitate communications about the software. You will not normally make changes to any member in the SOURCE PDS. However, you **will** make changes to members in the USOURCE PDS. These changes are described in the appropriate User Manuals for CPExpert components.

The names of the members of the PDS generally denote the component with which they are associated. The component is identified by the first three characters of the PDS member (e.g., WLMxxxxx, DASxxxxx, CICxxxxx, etc.).

Members used by more than one component are prefixed with the characters "GEN" (for example, GENMODEL is a general module used to compute the probability of finding a server busy, using Erlang C formula ( $m/m/c$ ), and can be used by any component of CPExpert).

There are two special members (titled **AAAAAAAA** and **ZZZZZZZZ**) located in SOURCE. These two members describe the most recent release level (and PTF, if appropriate) of CPExpert code contained in SOURCE.



## Chapter 2: Allocate space for CPEXpert data sets.

The second step is to allocate space for SAS libraries maintained by CPEXpert. CPEXpert maintains SAS data sets to describe SYS1.PARMLIB members, to describe problems that are discovered, and to contain historical information. These data sets contain the results from each produced by each CPEXpert component processing system measurement data.

Exhibit 2-2 illustrates the JCL required to allocate space for CPEXpert SAS libraries. SAS will automatically generate optimal DCB parameters.

---

//jobname	JOB	job card information
//STEP01	EXEC	IEFBR14
//CPEDATA	DD	DSN=prefix.CPEXPERT.CPEDATA,DISP=(,CATLG),
//		VOL=SER=xxxxxx,UNIT=3380,SPACE=(CYL,(10,1))
//HISTORY	DD	DSN=prefix.CPEXPERT.HISTORY,DISP=(,CATLG),
//		VOL=SER=xxxxxx,UNIT=3380,SPACE=(CYL,(5,1))

---

### JOB CONTROL LANGUAGE TO ALLOCATE DASD SPACE

#### EXHIBIT 2-2



## Section 3: Providing general guidance

---

This section describes how to modify the General Environment Controls section of the GENGUIDE Module. Most users will make changes to the General Environment Controls section only when installing CPEXpert, or if their environment changes.

**Most guidance parameters are optional.** Only the first parameter (the PDB type) and potentially the second parameter (the MVSVER to specify the MVS version) are required. All other parameters are provided in response to the unique requirements of individual users of CPEXpert.

You should review the options to see if any apply to your organization.

Exhibit 3-1 illustrates the General Environment Controls section of the GENGUIDE Module. The following discussion provides information about each of the parameters in the General Environment Controls section:

```

*****;
* SET GENERAL ENVIRONMENT CONTROLS.
*****;

%LET PDB          = MXG          ; * INPUT IS FROM MXG PERFORMANCE DATA BASE          ;
%LET MVSVER       = SP6.10      ; * MVS VERSION (SPECIFY SP7.n FOR z/OS              ;
%LET PDBLIB       = PDBLIB      ; * SAS VARIABLE FOR THE PDB LIBRARY              ;
%LET TYPE30_1     = N           ; * TYPE 30 JOB INITIATION (Y= AVAILABLE)              ;
%LET TYPE30_4     = N           ; * TYPE 30 STEP TERMINATION (Y = AVAILABLE)          ;
%LET TYPE30_5     = N           ; * TYPE 30 JOB TERMINATION (Y = AVAILABLE)          ;
%LET TYPE30_V     = N           ; * TYPE 30 INTERVAL ACCOUNTING (Y= AVAILABLE)        ;
%LET TYPE30DD     = N           ; * MXG TYPE30 MOD. (Y = AVAILABLE)                  ;
%LET TYPE42DS     = N           ; * MXG TYPE42DS      (Y = AVAILABLE)                  ;
%LET TYPE64       = N           ; * MXG TYPE64        (Y = AVAILABLE)                  ;
%LET SMFTYP88     = N           ; * TYPE 88 RECORDS AVAILABLE IN MXG?                  ;
%LET PAGESIZE     = 65          ; * STANDARD PAGE SIZE FOR PRINTED OUTPUT          ;
%LET PRINT        = PRINT       ; * USE STANDARD SAS PRINT OUTPUT FILE              ;
%LET PRSM         = N           ; * YOUR SYSTEM OPERATES UNDER PR/SM, MDF, MLPF      ;
%LET CPUPAGE      = .165        ; * APPROXIMATE CPU SERVICE UNITS PER PAGE-IN        ;
%LET CPUEXCP      = .725        ; * APPROXIMATE CPU SERVICE UNITS PER EXCP          ;
%LET CPUSWAP      = 7.59        ; * APPROXIMATE CPU SERVICE UNITS PER SWAP          ;
%LET CYCLE        = 01          ; * MICS PROCESSING CYCLE (01 = CYCLE 01)            ;
%LET MICSLEV      = RMF6380     ; * MICS PRODUCT CHANGE LEVEL                  ;
%LET MICSDERV     = N           ; * MICS DERIVATION MACROS                      ;
%LET HARLIB       = &PDBLIB     ; * MICS HAR INFORMATION AREA                      ;
%LET SCPLIB       = &PDBLIB     ; * MICS SCP INFORMATION AREA                      ;
%LET BATLIB       = &PDBLIB     ; * MICS BAT INFORMATION AREA                      ;
%LET DB2LIB       = &PDBLIB     ; * MICS DB2 INFORMATION AREA                      ;
%LET MICJOB       = Y           ; * MICS BATJOB AVAILABLE?                      ;
%LET MIC_TS       = Y           ; * MICS BAT_TS AVAILABLE?                      ;
%LET MIC_ST       = Y           ; * MICS BAT_ST AVAILABLE                      ;

```

## SAMPLE DISPLAY OF CPEXPRT.USOURCE(GENGUIDE)

### EXHIBIT 3-1

```

%LET MXG30_1 =           ; * MXG TYPE30_1 FULL NAME           ;
%LET MXG30_4 =           ; * MXG TYPE30_4 FULL NAME           ;
%LET MXG30_5 =           ; * MXG TYPE30_4 FULL NAME           ;
%LET MXG30_V =           ; * MXG TYPE30_V FULL NAME           ;
%LET MXG42DS =           ; * MXG TYPE42DS FULL NAME           ;
%LET MXG64  =           ; * MXG TYPE64 FULL NAME              ;
%LET MXG70  =           ; * MXG TYPE70 FULL NAME              ;
%LET MXG70PR =           ; * MXG TYPE70PR FULL NAME           ;
%LET MXG71  =           ; * MXG TYPE71 FULL NAME              ;
%LET MXG72  =           ; * MXG TYPE72 FULL NAME              ;
%LET MXG72GO =           ; * MXG TYPE72GO FULL NAME           ;
%LET MXG72SC =           ; * MXG TYPE72SC FULL NAME           ;
%LET MXG72DL =           ; * MXG TYPE72DL FULL NAME           ;
%LET MXG74  =           ; * MXG TYPE74 FULL NAME              ;
%LET MXG74CF =           ; * MXG TYPE74CF FULL NAME           ;
%LET MXG74ME =           ; * MXG TYPE74ME FULL NAME           ;
%LET MXG74PA =           ; * MXG TYPE74PA FULL NAME           ;
%LET MXG74ST =           ; * MXG TYPE74ST FULL NAME           ;
%LET MXG74SY =           ; * MXG TYPE74SY FULL NAME           ;
%LET MXG75  =           ; * MXG TYPE75 FULL NAME              ;
%LET MXG78CF =           ; * MXG TYPE78CF FULL NAME           ;
%LET MXG88  =           ; * MXG TYPE88 FULL NAME              ;
%LET SASODS = N           ; * CONTROLS WHETHER SAS ODS IS USED ;
%LET PATH   =           ; * PATH FOR ODS OUTPUT              ;
%LET FRAME  = xxxFRAME    ; * GENERIC ODS FRAME NAME          ;
%LET CONTENTS = xxxxCONT  ; * GENERIC ODS CONTENTS NAME       ;
%LET BODY   = xxxxBODY    ; * GENERIC ODS BODY NAME           ;
%LET LINKPDF =           ; * LINK TO CPEXPRT DOCUMENTATION    ;
%LET STYLE  =           ; * ODS HTML STYLE OPTION             ;
%LET PDFODS = N           ; * CONTROLS WHETHER SAS PDF IS USED ;
%LET PDFFILE = filename   ; * DEFINES THE SAS PDF OUTPUT FILE  ;
%LET TRANTAB = EBCDIC     ; * CONTROLS CHARACTER FORMAT OF HTML OUTPUT ;
%LET URL    = N           ; * CONTROLS .HTM IN SAS ODS FRAME OUTPUT ;
*****;

```

## SAMPLE DISPLAY OF CPEXPRT.USOURCE(GENGUIDE)

### EXHIBIT 3-1 (Continued)

## Chapter 1: Type of performance data base

**This is a required parameter.** The **%LET PDB = MXG** statement defines the type of performance data base CPExpert is to analyze.

Change the specification to **%LET PDB = MICS** to process a MICS performance data base.

## Chapter 2: MVS version

This is a required parameter **only** if you are executing the CPExpert **SRM Component**, CPExpert **MVS Component**, or the CPExpert **TSO Component**. This parameter is not required for other CPExpert Components.

The **%LET MVSVER = SPn.n** statement is a SAS macro statement to define the MVS version under which you are operating. This statement must be changed, if necessary, to reflect the MVS version under which your system operates.

**Use SP6.n for OS/390 releases of MVS, and use SP7.n for z/OS releases of MVS**, where “n” is the IBM Release Number of the version of MVS. For example, you should specify **%LET MVSVER = SP7.2** if you are operating under z/OS Version 1 Release 2.

## Chapter 3: PDBLIB JCL DD statement

The optional **%LET PDBLIB = PDBLIB** option was implemented at the request of a user who has the PDB organized into different SAS libraries - one library for each day. The user wanted to use SAS to automatically define the appropriate PDB library depending upon the day of week (e.g., Monday is MON.TYPE70, Tuesday is TUE.TYPE70, etc.).

The &PDBLIB macro variable was created and all references in the CPExpert code was changed from "PDBLIB.member" to "&PDBLIB..member". The user can prefix CPExpert with SAS code to determine the day of week and alter the global PDBLIB SAS macro variable accordingly. If you use this approach to refer to different performance data bases, you must remove the **%LET PDBLIB=PDBLIB** from the GENGUIDE member.

## Chapter 4: SMF Type 30 (Job Initiation) records available

The optional **%LET TYPE30\_1 = N** statement tells CPExpert whether SMF Type 30 (Job Initiation) records are available for analysis. If you collect the SMF Type 30 (Job Initiation) records, CPExpert can perform more analysis into potential performance problems.

This statement should be changed to **%LET TYPE30\_1 = Y** if you collect SMF Type 30 (Job Initiation) records for the system(s) being analyzed.

## Chapter 5: SMF Type 30 (Job Step) records available

The optional **%LET TYPE30\_4 = N** statement tells CPEXpert whether SMF Type 30 (Job Step) records are available for analysis. The SMF Type 30 (Job Step) records are the SMF Type 30 (Subtype 4) records.

This statement should be changed to **%LET TYPE30\_4 = Y** if you collect SMF Type 30 (Workload) records for the system(s) being analyzed. CPEXpert can perform more analysis into potential performance problems if you collect the SMF Type 30 (Job Step) records.

## Chapter 6: SMF Type 30 (Job) records available

The optional **%LET TYPE30\_5 = N** statement tells CPEXpert whether SMF Type 30 (Job) records are available for analysis. The SMF Type 30 (Job) records are the SMF Type 30 (Subtype 5) records.

This statement should be changed to **%LET TYPE30\_5 = Y** if you collect SMF Type 30 (Workload) records for the system(s) being analyzed. CPEXpert can perform more analysis into potential performance problems if you collect the SMF Type 30 (Job) records.

## Chapter 7: SMF Type 30 (Interval) records available

The optional **%LET TYPE30\_V = N** statement tells CPEXpert whether SMF Type 30 (Interval) records are available for analysis. If you collect the SMF Type 30 (Interval) records, CPEXpert can perform more analysis into potential performance problems.

This statement should be changed to **%LET TYPE30\_V = Y** if you collect SMF Type 30 (Interval) records for the system(s) being analyzed.

## Chapter 8: SMF Type 30 modification to MXG or MICS installed

The optional **%LET TYPE30DD = N** statement tells CPEXpert whether you have installed the CPEXpert modification to MXG or MICS code to collect SMF Type 30 (Data Definition) information.

The modification included with CPEXpert is quite simple. While MXG or MICS is processing SMF Type 30 records, CPEXpert records only the essential information required to identify the job name, the step name, service class (for Goal Mode) or the

performance group number (for Compatibility Mode), the devices referenced and their Start I/O counts and connect times, and the beginning and end time of the measurement. This information is sufficiently concise that less than 25 cylinders of DASD are required to hold the information for all job steps executed in a relatively large installation.

This modification allows CPEXpert to relate DASD performance information contained in SMF Type 74 records to specific service classes or performance groups, and to relate the Type 74 information to specific jobs or job steps.

The SMF Type 30 modification to MXG or MICS is available only with the DASD Component of CPEXpert, although other CPEXpert components use the information if the modification has been installed.

The TYPE30DD statement should be changed to **%LET TYPE30DD = Y** if you license the CPEXpert DASD Component, and if you have installed the CPEXpert modification to collect Type 30 (Data Definition) information for the system(s) being analyzed.

## Chapter 9: SMF Type 42 (Data Set) records available

The optional **%LET TYPE42DS = N** statement tells CPEXpert whether SMF Type 42 (Data Set) records are available for analysis. If you collect the SMF Type 42 (Data Set) records, CPEXpert can perform more analysis into potential performance problems. The SMF Type 42 (Data Set) records are highly recommended if you want the DASD Component to analyze VSAM performance problems.

This statement should be changed to **%LET TYPE42DS = Y** if you collect SMF Type 42 (Data Set) records for the system(s) being analyzed.

## Chapter 10: SMF Type 64 (VSAM Statistics) records available

The optional **%LET TYPE64 = N** statement tells CPEXpert whether SMF Type 64 (VSAM Statistics) records are available for analysis. If you collect the SMF Type 64 (VSAM Statistics) records, CPEXpert can perform analysis of potential performance problems with VSAM data sets.

This statement should be changed to **%LET TYPE64 = Y** if you collect SMF Type 64 (VSAM Statistics) records for the system(s) being analyzed.

## Chapter 11: SMF Type 88 (System Logger) records available

The optional **%LET TYPE88 = N** statement tells CPEXpert whether SMF Type 88 (System Logger) records are available for analysis. If you collect the SMF Type 88 (System



Logger) records, CPExpert can perform more analysis into potential performance problems.

This statement should be changed to **%LET TYPE88 = Y** if you collect SMF Type 88 (System Logger) records for the system(s) being analyzed.

## Chapter 12: Page size for printer output

The optional **%LET PAGESIZE = 65** statement defines the standard page size. CPExpert uses this information to make sure that the narrative associated with each rule is contained within a single page.

## Chapter 13: SAS printer output destination

The optional **%LET PRINT = PRINT** statement defines the standard SAS print output. The statement may be re-coded to define an output destination other than the standard print output.

## Chapter 14: PR/SM, MDF, or MLPF

The optional **%LET PRSM = N** statement tells CPExpert whether your system operates under PR/SM (or MDF or MLPF). This statement should be changed to **%LET PRSM = Y** if you operate under PR/SM, MDF, or MLPF.

## Chapter 15: CPU service units (page-in) constant

The optional **%LET CPUPAGE = .165** statement tells CPExpert the approximate number of CPU service units that are required to service a page-in operation from auxiliary storage. This information is derived from the information provided by IBM in the MVS Initialization and Tuning Guide.

The default values for the PAGERT1 and PAGERT2 keywords which were in the IEAOPTxx member of SYS1.PARMLIB prior to MVS/ESA SP3.1<sup>1</sup> represent a paging rate which consumes five percent of the respective engine. Extrapolating the five percent allows a computation of the approximate number of service units required to perform a single page-in operation from auxiliary storage. Some analysts prefer to perform a multiple linear regression analysis (using page-in, page-out, swap, EXCP, etc.) to account for the

---

<sup>1</sup>While the PAGERT1 and PAGERT2 keywords have not been available in IEAOPTxx since MVS/ESA Version 3.1, the corresponding constants are located in the MVS internal constants module IRARMCNS.

uncaptured CPU time. If you use such a method and derive different CPU service units for page-in operations, you can use the **CPUPAGE** variable to guide CPEXpert with your unique values.

## Chapter 16: CPU service units (EXCP) constant

The optional **%LET CPUEXCP = .725** statement tells CPEXpert the approximate number of CPU service units that are required to service an I/O operation. This information is derived from the information published in professional journals (e.g., the Proceedings from the CMG International conferences). The published information was based upon multiple linear regression analysis (using page-in, page-out, swap, EXCP, etc.) to account for the uncaptured CPU time. If you use such a method and derive different CPU service units for I/O operations, you can use the **CPUEXCP** variable to guide CPEXpert with your unique values.

## Chapter 17: CPU service units (swap) constant

The optional **%LET CPUSWAP = 7.59** statement tells CPEXpert the approximate number of CPU service units that are required to service a swap operation from auxiliary storage. This information is derived from the information published in professional journals (e.g., the Proceedings from the CMG International conferences).

The published information was based upon multiple linear regression analysis (using page-in, page-out, swap, EXCP, etc.) to account for the uncaptured CPU time. If you use such a method and derive different CPU service units for I/O operations, you can use the **CPUSWAP** variable to guide CPEXpert with your unique values.

## Chapter 18: MICS processing cycle

**This optional parameter applies only to users with a MICS performance data base.**

The **%LET CYCLE = 01** statement allows you to specify a MICS processing cycle other than the most recently-created cycle.

## Chapter 19: MICS product change level

**This optional parameter applies only to users with a MICS performance data base. Although the parameter is optional, all MICS users should verify the product change level of MICS before running CPEXpert!**

The **%LET MICSLEV =RMF6380** statement allows you to specify the MICS product change level that created the performance data base to be analyzed by CPExpert. CPExpert has identified a variety of errors with MICS, particularly in OS/390 Goal Mode data. Depending on the MICS product change level, the MICS developers have corrected the errors. The CPExpert code can either suppress analysis or (in some cases) apply corrections to the MICS variables. However, CPExpert must be able to identify which MICS product change level created the performance data base being analyzed.

MICS retains a SAS variable titled "MICSVER" in the MICS data sets. Unfortunately, the MICSVER variable contains "60" and is meaningless with regard identifying the MICS product change level. The CA-MICS product development staff is considering adding a new variable to MICS (perhaps in the HARCPU data set) that will be updated to reflect the MICS product change level. If this variable is added, CPExpert can automatically determine the MICS product change level and can decide which corrections (if any) to apply.

Absent the variable in a MICS data set to identify the product change level, the CPExpert user must verify the MICS product change level that created the data in the performance data base.

## Chapter 20: MICS variable derivation option

**This optional parameter applies only to users with a MICS performance data base.**

MICS derives many variables from basic information during its processing<sup>2</sup>. A MICS user has the option of (1) having MICS compute these variables as MICS builds its performance data base and storing the derived variables in appropriate data sets in the MICS data base, or (2) computing the variables each time the particular MICS data set is referenced. With the first option, the variables are computed only once, but storing the variables requires DASD space. With the second option, the variables are not stored on DASD, but processing time is required to compute the variables whenever they are referenced.

The MICSDERV statement allows you to specify whether the MICS derivation macros should be invoked by CPExpert as it processes a MICS performance data base. The default (**%LET MICSDERV = N**) causes CPExpert not to include the MICS derivation macros. Specify **%LET MICSDERV = Y**; if you wish CPExpert to include the MICS derivation macros whenever it references a MICS performance data base.

**WARNING: If you wish CPExpert to include the MICS derivation macros, you must include a SASAUTOS DD statement in your Job Control Language, pointing to the**

---

<sup>2</sup>For example, MICS computes averages for many variables.

**MICS MACAUTOS macro library.** Check with your MICS system administrator if you do not know the data set name of the MICS MACAUTOS macro library.

## Chapter 21: MICS JCL DD statement options

**These optional parameters apply only to users with a MICS performance data base.**

MICS divides its various files into logical "information areas" which can be contained in a single SAS library within a MICS data base unit or complex, or which can be in individual SAS libraries (perhaps on different volumes). Some installations have split the MICS information areas into separate SAS libraries. This action normally has been taken because of the size of the SAS files contained in the information areas.

CPEXpert normally uses the DD name PDBLIB (or whatever name you have specified if you modify the %LET PDBLIB = PDBLIB statement) to describe the SAS library in which the MICS information areas reside. You can direct CPEXpert to use a different SAS library for selected MICS information areas if you wish CPEXpert to use a different SAS library for any particular information area.

- The **%LET HARLIB = &PDBLIB** statement allows CPEXpert to reference the HAR information area wherever it may be placed.

You do not need to use the %LET HARLIB = statement if MICS has been installed with the HAR, SCP, BAT, and CIC information areas in a single data base unit or complex. The default statement in GENGUIDE (%LET HARLIB = &PDBLIB) causes CPEXpert to reference the HAR information area files in the SAS library pointed to by the PDBLIB DD statement.

However, you can uniquely define the SAS library containing the MICS HAR information area if your HAR information area is not in the SAS library pointed to by the PDBLIB DD JCL statement. You might, for example, specify **%LET HARLIB = HARLIB** to cause CPEXpert to use the SAS library pointed to by the HARLIB DD JCL statement whenever CPEXpert requires files in the HAR information area.

CPEXpert establishes a default null value for the HARLIB macro variable, and tests this variable after including CPEXPRT.USOURCE(GENGUIDE). If the HARLIB variable is null after including GENGUIDE, CPEXpert will set the HARLIB variable to the value of the PDBLIB macro variable. The point of this is that you do not have to worry about including the HARLIB variable in GENGUIDE if you are an existing user of CPEXpert.

- The **%LET SCPLIB = &PDBLIB** statement applies only to users with a MICS performance data base. See the above discussion related to the HARLIB variable for an explanation of the SCPLIB variable.

- The **%LET BATLIB = &PDBLIB** statement applies only to users with a MICS performance data base. See the above discussion related to the HARLIB variable for an explanation of the BATLIB variable.
- The **%LET DB2LIB = &PDBLIB** statement applies only to users with a MICS performance data base. See the above discussion related to the HARLIB variable for an explanation of the DB2LIB variable.

## Chapter 22: MICS MICJOB, MIC\_TS, and MIC\_ST variables

**These optional parameters apply only to users with a MICS performance data base.**

MICS places batch job, TSO, and started tasks information from the SMF Type 30 records into the MICS BATJOB, BAT\_TS, and BAT\_ST files, respectively. Not all sites retain these files, but some CPEXpert code requires data in these files. The MICJOB, MIC\_TS, and MIC\_ST allows CPEXpert to function properly in these files are not available.

- The **%LET MICJOB = Y** variable controls whether CPEXpert should expect that the MICS BATJOB file is available. If MICJOB=Y is specified, CPEXpert will expect the BATJOB file to be in the SAS library pointed to by the BATLIB statement. If MICJOB=N is specified, CPEXpert will suppress code that requires the MICS BATJOB file.
- The **%LET MIC\_TS = Y** variable controls whether CPEXpert should expect that the MICS BAT\_TS file is available. If MIC\_TS=Y is specified, CPEXpert will expect the BAT\_TS file to be in the SAS library pointed to by the BATLIB statement. If MICJOB=N is specified, CPEXpert will suppress code that requires the MICS BAT\_TS file.
- The **%LET BAT\_ST = Y** variable controls whether CPEXpert should expect that the MICS BATJOB file is available. If BAT\_ST=Y is specified, CPEXpert will expect the BATJOB file to be in the SAS library pointed to by the BATLIB statement. If BAT\_ST=N is specified, CPEXpert will suppress code that requires the MICS BAT\_ST file.

Please note that some CPEXpert components provide unique guidance capability for the MICS files used by the specific component. For example, the CPEXpert DB2 Component provides flexibility with respect to the MICS DB2xxx files. Please refer to the User Manual for the relevant CPEXpert component for additional information.

## Chapter 23: MXG JCL DD statement options

**These optional parameters apply only to users with a MXG performance data base.** These parameters are used **only** if your MXG data sets are placed into different SAS libraries (perhaps residing on different DASD volumes).

CPExpert normally uses the DD name PDBLIB (or whatever name you have specified if you modify the %LET PDBLIB = PDBLIB statement) to describe the SAS library in which the MXG data sets reside.

You can direct CPExpert to use a different SAS library for selected data sets if you wish CPExpert to use a different SAS library for any particular data set. This is accomplished by changing the appropriate %LET MXGxx statement to refer to the **full name of the MXG data set**, including the SAS library DD name.

For example, suppose you have placed the SMF Type 74 information processed by MXG into a SAS data set titled "MYTYPE74" and you wish to refer to this SAS data set by the DD statement named "MYLIB74". You would code the following to cause CPExpert to refer to your Type 74 information as MYLIB74.MYTYPE74:

```
%LET MXG74 = MYLIB74.MYTYPE74;
```

CPExpert establishes a default null value for each MXGxx macro variable. CPExpert tests this macro variable after including CPEXPERT.USOURCE(GENGUIDE). If any MXGxx variable is null after including GENGUIDE, CPExpert will set the variable to the value of the &PDBLIB macro variable concatenated with "TYPExx" for the appropriate MXG TYPExx data set. The point of this is that you do not have to worry about including any of the MXGxx variables in GENGUIDE if you are an existing user of CPExpert.

Additionally, you do not have to specify the name of all MXG TYPExx data sets. You need to specify only the data sets residing in a SAS library other PDBLIB.

Please note that some CPExpert components provide unique guidance capability for the MXG files used by the specific component. For example, the CPExpert CICS Component provides flexibility with respect to the MXG CICxxx files. Please refer to the User Manual for the relevant CPExpert component for additional information.

## Chapter 24: SAS Output Delivery System

Output from CPExpert is created using Basic SAS statements. This Basic SAS output is designed for a standard SAS printer (line) format. With SAS Release 8, SAS users can use the SAS Output Delivery System to create output that is formatted in Hypertext Markup Language (HTML). This output can be browsed with Internet Explorer, Netscape, or any other browser that fully supports the HTML 3.2 tag set.

The CPEXpert WLM Component, DB2 Component, CICS Component, and DASD Component support the SAS ODS features<sup>3</sup>.

The below options can be placed in USOURCE(GENGUIDE) so they apply to all CPEXpert components. Alternatively, the options can be placed in USOURCE(xxxGUIDE), where “xxx” is the suffix of a particular CPEXpert component to which the specifications apply.

## Chapter 24.1: SAS Output Delivery System - SASODS variable

The **SASODS** guidance variable is applicable only if you wish to exercise the SAS ODS features of CPEXpert. The SASODS variable tells CPEXpert to implement SAS ODS commands to open HTML files and to direct output from CPEXpert to those HTML files **in addition to** the output that goes to the standard SAS printer destination.

The default value for the SASODS variable is “N”, indicating that the SAS ODS feature should not be used by CPEXpert. CPEXpert can be directed to implement the SAS ODS feature<sup>4</sup> by specifying **%LET SASODS=Y;** in USOURCE(xxxGUIDE), where “xxx” is the WLM, DB2, CICS, or DAS to indicate the WLM Component, DB2 Component, CICS Component, or DASD Component. Alternatively, the **%LET SASODS=Y;** statement can be placed in USOURCE(GENGUIDE) so all CPEXpert components will create SAS ODS output.

In addition to specifying **%LET SASODS=Y;** in USOURCE(xxxGUIDE), you must take the following steps to use the SAS ODS feature:

- Allocate a partitioned data set where SAS will place the HTML output from CPEXpert components. Each CPEXpert component will “open” and “close” the HTML destinations, using FRAME, CONTENTS, and BODY file names (PDSE member names) that you have defined to the particular component. The definitions of these file names are described below.

The SAS Institute gives the following example for allocating the partitioned data set under SAS (of course, this data set allocation can be implemented using a JCL DD statement):

---

<sup>3</sup>You normally would specify the SAS ODS guidance variables in the guidance member for the respective CPEXpert component. For example, you normally would specify guidance for the WLM Component in USOURCE(WLMGUIDE). The discussion is included in this CPEXpert *Installation Guide* document because most new CPEXpert users would appreciate knowing the process involved with exercising the SAS ODS features as a part of installing CPEXpert.

<sup>4</sup>The SASODS guidance variable and other variables related to the SAS ODS feature are ignored unless you are executing at SAS Version 8.

```
/* Allocate a PDSE for the HTML Output. */  
filename pdsehtml '.example.htm'  
           dsntype=library dsorg=po  
           disp=(new, catlg, delete);
```

**NOTE:** If your organization specifies default DCB attributes, please make sure that you specify RECFM=VB, LRECL=8196,BLKSIZE=27998 and DSORG=PO in JCL.

- Identify to CPEXpert the data set that was created as shown above (the example uses “pdsehtml” as the data set name, but you can use another name). This identification is accomplished using the **PATH** statement described in Chapter 1.2 below. **This is a required step if you exercise the SAS ODS feature.**
- Specify **%LET SASODS=Y;** in USOURCE(GENGUIDE).
- Optionally specify a value for the **FRAME** variable, described in Chapter 24.3 below.
- Optionally specify a value for the **CONTENTS** variable, described in Chapter 24.4 below.
- Optionally specify a value for the **BODY** variable, described in Chapter 24.5 below.
- Optionally specify a value for the **LINKPDF** variable, described in Chapter 24.6 below.
- Optionally specify a value for the **STYLE** variable, described in Chapter 24.7 below.
- Optionally specify a value for the **PDFODS** variable, described in Chapter 24.8 below.
- Optionally specify a value for the **PDFFILE** variable, described in Chapter 24.9 below (**REQUIRED** if PDFODS is used).
- Optionally specify a value for the **TRANTAB** variable, described in Chapter 24.10 below.
- Optionally specify a value for the **URL** variable, described in Chapter 24.11 below.

Some users wish to access the SAS ODS output created by CPEXpert on their IBM mainframe in a PDSE file, some users wish to access the SAS ODS output in an HFS directory, some users wish to download the SAS ODS output to a web server, some users wish to download the SAS ODS output to a PC, and some users will create and access the SAS ODS output on a PC under Windows. The below table shows the various settings



that should be specified, depending on where the SAS ODS output was created and where it will be accessed.

	System where SAS ODS will be accessed				
CPEXpert feature	IBM OS/390 PDSE	HFS Directory	Download to Web server	Download to Windows	Created under Windows
FRAME file	filename	filename	filename	filename	filename
CONTENTS file	filename	filename.htm	filename	filename	filename
BODY file	filename	filename.htm	filename	filename	filename
URL	N	N	Y	Y	Y
Comments			Transfer to web server as binary file. Add .htm to file names during download. Translate to ASCII during download.	Add .htm to file names during download. Translate to ASCII during download.	

## Chapter 24.2: SAS Output Delivery System - PATH variable

The PATH variable is applicable only if you exercise the SAS ODS features, however **this is a required variable if you use the SAS ODS feature; there is no default specification for this variable.**

The PATH variable specifies the file name (or the DDNAME) of the external file where all HTML output produced by CPEXpert will be placed. If you use a DDNAME, the file specification would be the same as the DDNAME name used when creating the partitioned data set as described above. For example, you might have used "CPEHTML" as the

DDNAME name, and the actual data set name might be prefix.CPEXPERT.HTM. In this case<sup>5</sup>, the PATH variable would be specified as **%LET PATH=CPEHTML;**

## Chapter 24.3: SAS Output Delivery System - FRAME variable

The FRAME variable is applicable only if you exercise the SAS ODS features. The FRAME variable identifies the file (PDSE member) that integrates the HTML table of contents and the HTML body file. A table of contents and the body file are displayed when the frame file is opened using a standard web browser. The default value used by CPExpert for the FRAME variable is **xxxFRAME**, where “xxx” is WLM, DB2, CICS, or DAS for the WLM Component, DB2 Component, CICS Component, or DASD Component, respectively. You can alter this default by changing the value of the FRAME variable. This name is where you would point your browser to examine output created by the particular CPExpert component.

## Chapter 24.4: SAS Output Delivery System - CONTENTS variable

The CONTENTS variable is applicable only if you exercise the SAS ODS features. The CONTENTS variable identifies the file (PDSE member) that contains a table of contents to the HTML output from the CPExpert component. The contents file links to the body file, which holds the actual output from the CPExpert component. The default value used by the CPExpert for the BODY variable is **xxxCONT**, where “xxx” is WLM, DB2, CICS, or DASD for the WLM Component, DB2 Component, CICS Component, or DASD Component, respectively. You can alter this default by changing the value of the CONTENTS variable.

## Chapter 24.5: SAS Output Delivery System - BODY variable

The BODY variable is applicable only if you exercise the SAS ODS features. The BODY variable identifies the file (PDSE member) that contains the HTML output from the CPExpert component. The default value used by CPExpert for the BODY variable is **xxxBODY**, where “xxx” is WLM, DB2, CICS, or DASD for the WLM Component, DB2 Component, CICS Component, or DASD Component, respectively. You can alter this default by changing the value of the BODY variable.

---

<sup>5</sup>The PATH specification would be a full directory identification if operating under Windows on a PC. For example, you might specify **%LET PATH=C:\CPEXPERT\OUTPUT\HTML;** and CPExpert would place all HTML output in this HTML directory.

## Chapter 24.6: SAS Output Delivery System - LINKPDF variable

The LNKPDF variable is applicable only if you exercise the SAS ODS features, and if you wish to have the HTML rules link to rule descriptions in the CPExpert documentation.

The HTML output in the BODY file produced by the CPExpert component can **optionally** contain links to the CPExpert Rule Description documentation<sup>6</sup>. You can use a browser to browse the FRAME (or the BODY) file and if you wish to reference the CPExpert documentation associated with a particular finding, you can simply click on the link produced in the Rule Heading line.

The default value used by the CPExpert component for the LINKPDF variable is blank (null), indicating that links should not be placed in the body of the SAS ODS output. You can alter this default by changing the value of the LINKPDF variable in one of two ways:

- Specify **%LET LINKPDF = URL address;** in USOURCE(xxxGUIDE). This specification will cause CPExpert to create links in the Rule Description title line of every rule that is produced by the CPExpert component. With this specification, the CPExpert Rule Descriptions can be in a file specified by the **URL address** argument. For example, suppose that Rule CIC107 were produced by the CICS Component. If you specified **%LET LINKPDF = http://www.cpexpert.com/;**, CPExpert would generate a link to <http://www.cpexpert.com/CIC107.pdf><sup>7</sup>.
- Specify **%LET LINKPDF = Y;** in USOURCE(xxxGUIDE). This specification will cause CPExpert to create links in the Rule Description title line of every rule that is produced by the CPExpert component. With this specification, the CPExpert Rule Descriptions must be in the same partitioned data set as is the xxxBODY, where “xxx” is WLM, DB2, CICS, or DASD for the WLM Component, DB2 Component, CICS Component, or DASD Component, respectively. These CPExpert Rule Descriptions could have been loaded to the partitioned data set that was defined as explained earlier.

**The CD issued with each release of CPExpert contains directories that have individual Rule Descriptions for the CPExpert components that you license. You should copy all “.pdf” Rule Descriptions for each CPExpert component that you license, from the CPExpert CD to the URL location for which you wish to have CPExpert place links.**

You should update the Rule Descriptions at the location specified by the URL when you receive a new CPExpert release, since many Rule Descriptions change with each new release.

---

<sup>6</sup>These links are not created by the SAS ODS software. The links are created by CPExpert after the standard CPExpert report has been produced. CPExpert modifies the output from SAS ODS to create the links.

<sup>7</sup>This example is for illustration purposes only. [www.cpexpert.com](http://www.cpexpert.com) does not contain CIC107.pdf. You should specify your own URL.

## Chapter 24.7: SAS Output Delivery System - STYLE variable

The STYLE variable is applicable only if you exercise the SAS ODS features. The SAS ODS STYLE option specifies the style definition to use in writing HTML files. The default value for the STYLE variable is %LET STYLE=;, which indicates that CPExpert should use the default SAS ODS STYLE option. You can specify %LET STYLE="your STYLE specification" if you wish to use a style definition that is unique to your organization. The STYLE variable can be used with either the HTML specification or the PDF specification.

## Chapter 24.8: SAS Output Delivery System - PDFODS variable

The PDFODS variable is applicable only if you exercise the SAS ODS features, and if you wish to have output directed to a PDF file. The default value for the PDFODS variable is %LET PDFODS=N;, which specifies that PDF output should not be created.

You can specify %LET PDFODS=Y; in USOURCE(GENGUIDE) if you wish SAS to create CPExpert output in PDF format. If you specify %LET PDFODS=Y;, you **MUST also specify a value for the PDFFILE variable described below.**

## Chapter 24.9: SAS Output Delivery System - PDFFILE variable

The PDFFILE variable is applicable only if you exercise the SAS ODS/PDF feature described above. The PDFFILE variable specifies the file name of the file in which SAS should place the PDF output. The PDFFILE name should specify either the DD name of the output, or a file described with a SAS FILENAME statement.

## Chapter 24.10: SAS Output Delivery System - TRANTAB variable

The TRANTAB variable is applicable only if you exercise the SAS ODS features. The default character format used by CPExpert for SAS ODS is ASCII. This format is suitable when the output is to be sent to an ASCII-based web server. You can override this character format to be EBCDIC if you do not plan to send the output to an ASCII-based web server, by specifying %LET TRANTAB=EBCDIC in USOURCE(GENGUIDE).

## Chapter 24.11: SAS Output Delivery System - URL variable

The URL variable is applicable only if you exercise the SAS ODS features. The default value for the URL variable is %LET URL=N;, which is applicable if you are going to access the SAS ODS output from CPExpert in IBM PDSE or in an HFS directory.

If you download the SAS ODS output to a web server, download the SAS ODS output to a PC, or are running CPExpert under WINDOWS, you must specify that a URL is to be used by SAS when it creates the SAS FRAME file. If **%LET URL=Y;** is specified in USOURCE(GENGUIDE), CPExpert will generate SAS code to create the URL specifications in the SAS FRAME file.



## Section 4: Files used by CPExpert

---

This section identifies the files used by the various components of CPExpert. For many sites, the required files will naturally be a part of the normal performance data base (PDB) retained at the site. A few sites might have to modify the SMF parameters so SMF records the required record types, or might have to modify the PDB update process so the required SAS data sets are retained in the PDB.

This section is divided into two chapters: (1) files required when processing a MXG performance data base and (2) files required when processing a NeuMICS performance data base.

## Chapter 1: Files required when processing a MXG PDB

This chapter identifies the files used when processing a MXG performance data base

Please note that some of the optional files are not routinely produced by MXG when executing the MXG BUILD PDB, and some files are not normally written to a performance data base by MXG.

- **CICS files.** If you use the CICS Component of CPExpert, you should make sure that the CICS files produced by MXG are copied to your performance data base. One way of accomplishing this is to use a PROC COPY after executing BUILD PDB:

```
%INCLUDE SOURCLIB(BUILD PDB);    /* BUILD PDB          */
PROC COPY IN=WORK OUT=PDB;        /* COPY CICS FILES TO PDB */
SELECT CICS;
```

- **TYPE42DS files.** If you use the DASD Component of CPExpert, and you wish CPExpert to analyze VSAM performance problems or you wish CPExpert to produce data set information related to devices, you must provide the TYPE42DS file to CPExpert. The TYPE42DS file is not normally produced by MXG. You must make sure that MXG produces the TYPE42DS file, and then must place that file into your performance data base. One way of accomplishing this is to include the MXG TYPE42 module, and use a PROC COPY after executing BUILD PDB:

```
%INCLUDE SOURCLIB(BUILD PDB);    /* BUILD PDB          */
%INCLUDE SOURCLIB(TYPE42);        /* PROCESS TYPE 42 RECORDS */
PROC COPY IN=WORK OUT=PDB;        /* COPY TYPE42DS FILES TO PDB */
SELECT TYPE42DS;
```

- **TYPE64 files.** If you use the DASD Component of CPExpert and you wish CPExpert to analyze VSAM performance problems, you must provide the TYPE64 and TYPE64X files to CPExpert. These files are not normally produced by MXG. You must make sure that MXG produces the TYPE64 and TYPE64X files, and then must place the files into your performance data base. One way of accomplishing this is to include the MXG TYPE64 module, and use a PROC COPY after executing BUILD PDB:

```
%INCLUDE SOURCLIB(BUILD PDB);    /* BUILD PDB          */
%INCLUDE SOURCLIB(TYPE64);        /* PROCESS TYPE 64 RECORDS */
PROC COPY IN=WORK OUT=PDB;        /* COPY TYPE64 FILES TO PDB */
SELECT TYPE64;
```



- **TYPE88 files.** If you use the WLM Component, the CICS Component (at CICS/TS), or the MVS Component of CPExpert, you should provide the TYPE88 file to CPExpert, so CPExpert can analyze potential system logger problems. The TYPE88 file is not normally produced by MXG. You must make sure that MXG produces the TYPE88 file, and then must place that file into your performance data base. One way of accomplishing this is to include the MXG TYPE88 module, and use a PROC COPY after executing BUILD PDB:

```
%INCLUDE SOURCLIB(BUILD PDB);    /* BUILD PDB                */
%INCLUDE SOURCLIB(TYPE88);       /* PROCESS TYPE 88 RECORDS  */
PROC COPY IN=WORK OUT=PDB;      /* COPY TYPE88 FILES TO PDB */
SELECT TYPE88;
```

## Chapter 1.1: Files used by the WLM Component

The following table identifies the files used by the WLM Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
Type 30	TYPE30_1	Optional.
Type 30	TYPE30_5	Optional
Type 30	TYPE30_V	Optional.
Type 30	DASD30DD	Optional: CPExpert modification to MXG.
Type 70	TYPE70	
Type 70	TYPE70PR	
Type 71	TYPE71	
Type 72	TYPE72GO	
Type 72	TYPE72DL	
Type 72	TYPE72SC	
Type 74	TYPE74	
Type 74	TYPE74CF	Optional: Required for coupling facility analysis.
Type 74	TYPE74ME	Optional: Required for coupling facility analysis.
Type 74	TYPE74PA	Optional: Required for coupling facility analysis.
Type 74	TYPE74ST	Optional: Required for coupling facility analysis.
Type 74	TYPE74SY	Optional: Required for coupling facility analysis.
Type 75	TYPE75	
Type 88	TYPE88	Optional: Required for system logger analysis

## Chapter 1.2: Files used by the DB2 Component

The following table identifies the files used by the DB2 Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
	DB2STATS	
Type 100 (Subtype 0)	DB2STATR	Applies only for DDF analysis
Type 100 (Subtype 1)	DB2STATB	
Type 100 (Subtype 1)	DB2GBPST	Applies only for data sharing analysis
Type 100 (Subtype 3)	DB2GBPAT	Applies only for data sharing analysis
Type 100 (Subtype 2)	DB2STAT2	
Type 101	DB2ACCT	Optional - some rules will be suppressed if not available
Type 102	T102S106	Optional, but highly recommended
Type 102	T102S172	Optional, but highly recommended
Type 102	T102S196	Optional, but highly recommended
Type 74	TYPE74ST	Optional, required for data sharing analysis

## Chapter 1.6: Files used by the CICS Component

The following table identifies the files used by the CICS Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
Type 110	CICAUTO	
Type 110(4)	CICCF6S6D	Optional; used with CFDT analysis
Type 110(4)	CICCF6S7D	Optional; used with CFDT analysis
Type 110(4)	CICCF6S8D	Optional; used with CFDT analysis
Type 110(4)	CICCF6S9D	Optional; used with CFDT analysis
Type 110	CICCONMR	
Type 110	CICCONSR	
Type 110	CICCONSS	
Type 110	CICDB2GL	Optional; used with CICS-DB2 Connect analysis
Type 110	CICDB2RE	Optional; used with CICS-DB2 Connect analysis
Type 110	CICDLIT	
Type 110	CICDQG	
Type 110	CICDS	
Type 110	CICDTB	
Type 110	CICFCR	
Type 110	CICJCR	N/A with CICS/TS
Type 110	CICLDR	
Type 110	CICLGR	N/A with CICS/TS
Type 110	CICLGS	Applicable with CICS/TS
Type 110	CICLSRFR	
Type 110	CICLSRR	
Type 110	CICNQG	
Type 110	CICRMG	
Type 110	CICSDG	

Type 110	CICSMDSA	
Type 110	CICSMT	
Type 110	CICTC	N/A with CICS 4.1 and above
Type 110	CICTCR	
Type 110	CICTCLR	N/A with CICS 4.1 and above
Type 110	CICTSQ	
Type 110	CICTSR	<b>N/A with CICS 4.1 and above</b>
Type 110	CICVT	
Type 110	CICXMC	Applicable with CICS 4.1 and above
Type 110	CICXMG	Applicable with CICS 4.1 and above
Type 110	CICXMR	Applicable with CICS 4.1 and above
Type 110(3)	CICXQ1	Applicable with CICS/TS
Type 110(3)	CICXQ2	Applicable with CICS/TS
Type 110(3)	CICXQ3	Applicable with CICS/TS
Type 88	TYPE88	Applicable with CICS/TS

## Chapter 1.5: Files used by the DASD Component

The following table identifies the files used by the DASD Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
Type 14/15	N/A	Optional: Required only if analyzing data set response performance. Generated by CPExpert.
Type 30	TYPE30_4	Optional.
Type 30	TYPE30_V	Optional.
Type 30	DASD30DD	Optional: CPExpert modification to MXG
Type 42	TYPE42DS	Optional: Required for VSAM and data set analysis
Type 64	TYPE64	Optional: Required for VSAM analysis
Type 64	TYPE64X	Optional: Required for VSAM analysis
Type 70	TYPE70	
Type 70	TYPE70PR	
Type 73	TYPE73	
Type 74	TYPE74	
Type 75	TYPE75	
Type 78	TYPE78CF	

## Chapter 1.3: Files used by the MVS Component (Compatibility Mode)

The following table identifies the files used by the MVS Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
Type 70	TYPE70	
Type 70	TYPE70PR	
Type 71	TYPE71	
Type 72	TYPE72	
Type 74	TYPE74	
Type 74	TYPE74CF	Optional: Required for coupling facility analysis.
Type 74	TYPE74ME	Optional: Required for coupling facility analysis.
Type 74	TYPE74PA	Optional: Required for coupling facility analysis.
Type 74	TYPE74ST	Optional: Required for coupling facility analysis.
Type 74	TYPE74SY	Optional: Required for coupling facility analysis.
Type 75	TYPE75	
Type 88	TYPE88	Optional: Required for system logger analysis

## Chapter 1.4: Files used by the TSO Component (Compatibility Mode)

The following table identifies the files used by the TSO Component of CPExpert when processing data in a MXG PDB:

SMF RECORD	MXG FILE	NOTES
Type 30	TYPE30_1	
Type 30	TYPE30_4	Optional.
Type 30	TYPE30_V	Optional.
Type 30	DASD30DD	Optional: CPExpert modification to MXG.
Type 70	TYPE70	
Type 70	TYPE70PR	
Type 71	TYPE71	
Type 72	TYPE72	
Type 74	TYPE74	
Type 75	TYPE75	

## Chapter 1.2: Files used by the SRM Component (Compatibility Mode)

No SMF or MXG files are used by the SRM Component. The SRM Component of CPExpert processes SYS1.PARMLIB members.



## Chapter 2: Files required when processing a NeuMICS PDB

This chapter identifies the files used when processing a NeuMICS performance data base

### Chapter 2.1: Files used by the WLM Component

The following table identifies the files used by the WLM Component of CPExpert when processing data in a NeuMICS PDB:

SMF RECORD	MICS FILE	NOTES
Type 30	BATJOB.cycle	Optional (but highly recommended).
Type 30	BAT_TS.cycle	Optional (but highly recommended).
Type 30	BAT_ST.cycle	Optional (but highly recommended).
Type 30	DASD30DD	Optional: CPExpert modification to MXG.
Type 70	HARCPU.cycle	
Type 70	HARLPC.cycle	
Type 71	SCPPAG.cycle	
Type 71	SCPSWP.cycle	
Type 72	WLMRGP.cycle	
Type 72	WLMSEA.cycle	
Type 72	WLM_EA.cycle	
Type 72	WLMSEC.cycle	
Type 72	WLMSED.cycle	
Type 72	WLMSEDE.cycle	
Type 72	WLMSDS.cycle	
Type 72	WLMSEM.cycle	
Type 72	WLMSVC.cycle	
Type 74	HARDVA.cycle	
Type 74	SCPXCA.cycle	Optional: Required for coupling facility analysis.
Type 74	SCPXCY.cycle	Optional: Required for coupling facility analysis.
Type 74	HARCFR.cycle	Optional: Required for coupling facility analysis.
Type 75	SCPPSD.cycle	

## Chapter 2.2: Files used by the DB2 Component

The following table identifies the files used by the DB2 Component of CPExpert when processing data in a NeuMICS PDB:

SMF RECORD	MICS FILE	NOTES
Type 100	DB2DSD	Applies only for DDF analysis
Type 100	DB2DDY	
Type 100	DB2DSY	
Type 100	DB2DSU	Applies only if DB2 accounting data is available
Type 100	DBPDSB	
Type 74	HARCFH	Optional, required for data sharing analysis
Type 74	HARCFR	Optional, required for data sharing analysis

## Chapter 2.3: Files used by the DASD Component

The following table identifies the files used by the DASD Component of CPExpert when processing data in a NeuMICS PDB:

SMF RECORD	MICS FILE	NOTES
Type 30		Optional.
Type 30		Optional.
Type 30	DASD30DD	Optional: CPExpert modification to MICS
Type 70	HARCPU.cycle	
Type 70	HARLPC.cycle	
Type 73	HARPCA.cycle	
Type 74	HARDVA.cycle	
Type 75	SCPPSD.cycle	
Type 78	SCPIOC.cycle	

## Chapter 2.4: Files used by the CICS Component

Beginning with CPExpert Release 12.1, the CICS Component of CPExpert no longer supports CICS performance analysis with NeuMICS.

**MXG is required for CPExpert to comprehensively process the CICS/ESA, CICS Transaction Server for OS/390, or CICS Transaction Server for z/OS data.** This is because the CICS information contained in the standard MICS data bases is insufficiently detailed for CPExpert to perform the detailed analysis of CICS performance problems. CPExpert previously attempted to perform **some** analysis of CICS data in a NeuNeuMICS data base, but most important areas could not be covered because the data is not normally retained in the NeuNeuMICS data sets. Consequently, CPExpert abandoned support of CICS performance analysis with a NeuNeuMICS performance data base.

## Chapter 2.5: Files used by the MVS Component (Compatibility Mode)

The following table identifies the files used by the MVS Component of CPEXpert when processing data in a NeuMICS PDB:

SMF RECORD	MICS FILE	NOTES
Type 70	HARCPU.cycle	
Type 70	HARLPC.cycle	
Type 71	SCPPAG.cycle	
Type 71	SCPSWP.cycle	
Type 72	SPCPPA.cycle	
Type 74	HARDVA.cycle	
Type 74	SCPXCA.cycle	Optional: Required for coupling facility analysis.
Type 74	SCPXCY.cycle	Optional: Required for coupling facility analysis.
Type 74	HARCFR.cycle	Optional: Required for coupling facility analysis.
Type 74	SCPPSD.cycle	Optional: Required for coupling facility analysis.
Type 74	HARCFR.cycle	Optional: Required for coupling facility analysis.
Type 75	SCPPSD.cycle	

## Chapter 2.6: Files used by the TSO Component (Compatibility Mode)

The following table identifies the files used by the TSO Component of CPEXpert when processing data in a NeuMICS PDB:

SMF RECORD	MICS FILE	NOTES
Type 70	HARCPU.cycle	
Type 70	HARLPC.cycle	
Type 71	SCPPAG.cycle	
Type 71	SCPSWP.cycle	
Type 72	SCPPPA.cycle	
Type 74	HARDVA.cycle	
Type 75	SCPPSD.cycle	

## Chapter 2.7: Files used by the SRM Component (Compatibility Mode)

No SMF or NeuMICS files are used by the SRM Component. The SRM Component of CPEXpert processes SYS1.PARMLIB members.

Section 4: Files used by CPExpert .....	1
Chapter 1: Files required when processing a MXG PDB .....	2
Chapter 1.1: Files used by the WLM Component .....	4
Chapter 1.2: Files used by the DB2 Component .....	5
Chapter 1.6: Files used by the CICS Component .....	6
Chapter 1.5: Files used by the DASD Component .....	8
Chapter 1.3: Files used by the MVS Component (Compatibility Mode) .....	9
Chapter 1.4: Files used by the TSO Component (Compatibility Mode) .....	10
Chapter 1.2: Files used by the SRM Component (Compatibility Mode) .....	10
Chapter 2: Files required when processing a NeuMICS PDB .....	11
Chapter 2.1: Files used by the WLM Component .....	11
Chapter 2.2: Files used by the DB2 Component .....	12
Chapter 2.3: Files used by the DASD Component .....	13
Chapter 2.4: Files used by the CICS Component .....	13
Chapter 2.5: Files used by the MVS Component (Compatibility Mode) .....	14
Chapter 2.6: Files used by the TSO Component (Compatibility Mode) .....	15
Chapter 2.7: Files used by the SRM Component (Compatibility Mode) .....	15

## Your turn:

This document has described how to install CPExpert.

We would appreciate receiving any comments you have regarding this document (style, content, clarity, etc.), or suggestions for improving the installation process. Please send your comments to:

Don Deese  
Computer Management Sciences, Inc.  
6076-D Franconia Road  
Alexandria, VA 22310  
**[www.cpexpert.com](http://www.cpexpert.com)**

---

Comments:

